CLAIMS

What is claimed is:

1. A method for providing transparent backup service from a primary node to a backup node for applications having objects running on the primary node, the method comprising:

defining a state for each application, the state having state parameters, and whereby the state can change with each user interaction with the application;

determining pooling arrangements for storing the states associated with each application in a session object associated with a particular session;

serializing the pooled states in the session object, with canonical representations being used for the various state parameters when available, and raw state parameter data being used otherwise; and

periodically replicating the session object over to the backup node, wherein upon fail-over the backup node can be instructed to parse through the serialized data and reconstruct the application objects according to the given state at fail-over, thereby providing a transparent backup for running the application.

2. The method according to Claim 1, wherein the applications are comprised of steps including composite steps, simple steps, interactive steps, and application steps.

3. The method according to Claim 2, wherein the pooling arrangement includes a global pool having at least a composite step pool, a simple step pool, an interactive step pool, and an application step pool.

4. The method according to Claim 1, wherein the canonical representations include logical references to an object identifier in a database associated with the nodes.

5. The method according to Claim 4, wherein the pooling arrangement includes hash lookup tables for efficiently storing and retrieving the pooled state information.

6. The method according to Claim 1, wherein the step of periodically replicating the session object occurs for each new user interaction with the nodes.

7. The method according to Claim 6, wherein user interaction occurs via a browser sending requests and receiving responses, and each browser session corresponds with a different session object.

8. The method according to Claim 1, wherein a web server device is associated with a plurality of nodes, and user requests are directed through the web server to the appropriate node.

9. The method according to Claim 1, wherein the step of periodically replicating the session object includes sending the session object from the primary node to the backup node over the associated network configuration.

10. The method according to Claim 1, wherein the step of periodically replicating the session object includes storing the session object on a shared storage medium for access by both the primary node and the backup node.

11. A distributed network server arrangement for providing transparent backup service from a primary node to a backup node for applications having objects running on the primary node, the server arrangement comprising:

a web server device for distributing requests from, and responses to, at least one browser device;

at least one active primary node for running the applications;

at least one backup node for running the applications in the event of fail-over of the primary node; and

a separate session object associated with each browser device, each session object being used for storing the serialized state representations of the applications with canonical representations being used as available for state representation data,

wherein the session object is periodically replicated over from the active primary node to the backup node, and the backup node can use the session object to reconstruct and run the application according to its state upon fail-over.

12. The distributed network server arrangement according to Claim 11, wherein the applications are comprised of steps including composite steps, simple steps, interactive steps, and application steps.

13. The distributed network server arrangement according to Claim 12, wherein the session object includes a pooling arrangement having a global pool with at least a composite step pool, a simple step pool, an interactive step pool, and an application step pool.

14. The distributed network server arrangement according to Claim 11, wherein the canonical representations include logical references to an object identifier in a database associated with the nodes.

15. The distributed network server arrangement according to Claim 13, wherein the pooling arrangement includes hash lookup tables for efficiently storing and retrieving the pooled state information.

16. The distributed network server arrangement according to Claim 11, wherein the periodic replication of the session object occurs for each new user interaction with the nodes.

17. The distributed network server arrangement according to Claim 16, wherein user interaction occurs via a browser sending requests and receiving responses, and each browser session corresponds with a different session object.

18. The distributed network server arrangement according to Claim 11, wherein the web server device is associated with a plurality of nodes, and user requests are directed through the web server to the appropriate node.

19. The distributed network server arrangement according to Claim 11, wherein the periodic replication of the session object includes sending the session object from the primary node to the backup node over the associated network configuration.

20. The distributed network server arrangement according to Claim 11, wherein the periodic replication of the session object includes storing the session object on a shared storage medium for access by the various nodes in the configuration.